SHRI MATA VAISHNO DEVI UNIVERSITY, KATRA
School of Computer Science & Engineering
B. Tech. (Computer Science & Engineering-Vth Sem.) Minor-I Examination (Odd) 2018-19

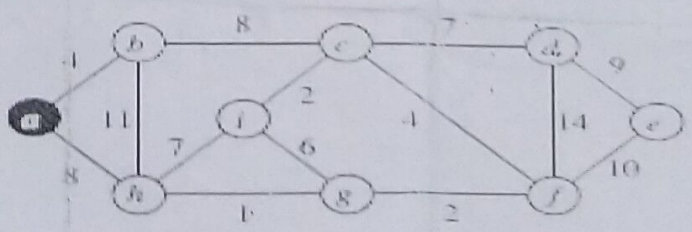
Entry No: 16bcs003 — Total Number of Pages: [02]
Date: 30.08.2018 — Total Number of Questions: [06]

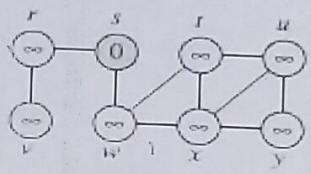
Course Title: Design & Analysis of Algorithms
Course Code: CSL 3032

Time Allowed: Time Allowed: 1 Hour and 30 Minutes — Max Marks: [20]

Instructions / NOTE

i. Attempt All Questions.
ii. Support your answer with neat freehand sketches/diagrams, wherever appropriate.
iii. Assume any missing data to suit the case / derivation / answer.
iv. Show every step while solving a numerical/analytical/design based question.
v. Maximum Marks assigned to each question are mentioned with the individual question.

| | Section - A | |
|---|---|---|
| Q1. | In case of finding Minimum Spanning Tree the edges are sorted in the following order of weight:<br>(i.) Increasing<br>(ii.) Decreasing<br>(iii.) Non-increasing<br>(iv.) Non-decreasing | [01] |
| Q2. | Which of the following Sorting Algorithm is based on Divide-and-Conquer Approach:<br>(i) Insertion<br>(ii) Merge<br>(iii) Both (i) & (ii)<br>(iv) None | [01] |
| Q3. | What is significance of Time and Space Complexity? Do you agree with the statement that "one specific type of complexity out of the two mentioned is sufficient to take in consideration"? Justify your answer with proper reasoning. | [03] |
| | Section - B | |
| Q4. | Considering vertex "a" as root vertex, find out the Minimum Spanning Tree and calculate weight of the same showing the execution of Prim's algorithm on the graph mentioned below:  | [05] |

| | | |
|---|---|---|
| Q5. | Write Kruskal's algorithm for Minimum Spanning Tree. Also write down the complexity of both the Prim's and Kruskal's Algorithm in all the cases. | [05] |
| Q6. | Write down the breadth-first-search algorithm and execute the same over the following graph considering vertex "s" as source vertex. Show each and every step clearly.  | [05] |

SHRI MATA VAISHNO DEVI UNIVERSITY, KATRA
School of Computer Science & Engineering
B. Tech. (Computer Science & Engineering-Vth Sem.) Minor-II Examination (Odd) 2018-19

Entry No: | | 1 | 6 | b | c | s | 0 | 0 | 3 |
Date: 15.10.2018

Total Number of Pages: [02]
Total Number of Questions: [09]

Course Title: Design & Analysis of Algorithms
Course Code: CSL 3032

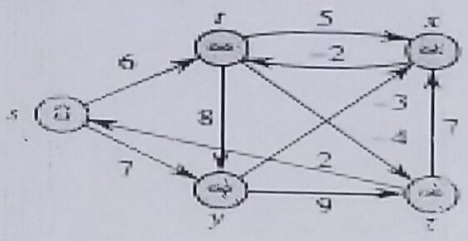
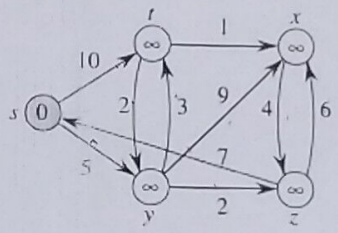
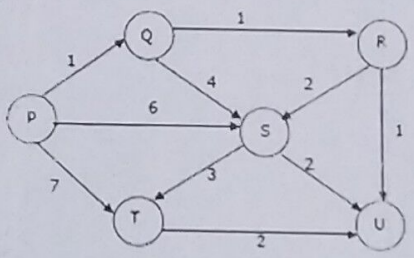
Time Allowed: Time Allowed: 1 Hour and 30 Minutes

Max Marks: [20]

Instructions / NOTE

i. Attempt All Questions.
ii. Support your answer with neat freehand sketches/diagrams, wherever appropriate.
iii. Assume any missing data to suit the case / derivation / answer.
iv. Show every step while solving a numerical/analytical/design based question.
v. Maximum Marks assigned to each question are mentioned with the individual question.

| Section - A | | |
|---|---|---|
| Q1. | What are negative weight cycles? Which algorithm can handle such situation? | [01] |
| Q2. | What is significance of Hamiltonian cycles? Point out positives and negatives of the same. | [01] |
| Q3. | What is the time complexity of Bellman-Ford single-source shortest path algorithm on a complete graph of n vertices in all the cases? | [01] |
| Q4. | Which of the following algorithm can be used to efficiently calculate single source shortest paths in a Directed Acyclic Graph?<br>(i) Dijkstra<br>(ii) Bellman-Ford<br>(ii) Topological Sort<br>(iv) Strongly Connected Component | [01] |
| Q5. | Given a directed graph where weight of every edge is same, we can efficiently find shortest path from a given source to destination using?<br>(i) Breadth First Traversal<br>(ii) Dijkstra's Shortest Path Algorithm<br>(iii) Neither Breadth First Traversal nor Dijkstra's algorithm can be used<br>(iv) Depth First Search | [01] |

| | Section - B | |
|---|---|---|
| Q6. | Considering the following graph, show the execution of Bellman-Ford algorithm. Also write the Bellman-Ford algorithm for single source shortest path problem.  | [05] |
| Q7. | For the following graph show the execution of Dijkstra's algorithm. Write down Dijkstra's algorithm and its time complexity in all the cases.  | [05] |
| Q8. | Write down the traveling-salesman problem algorithm and its time complexity in all the cases. | [03] |
| Q9. | Suppose we run Dijkstra's single source shortest-path algorithm on the following edge weighted directed graph with vertex P as the source. In what order do the nodes get included into the set of vertices for which the shortest path distances are finalized?  | [02] |

SHRI MATA VAISHNO DEVI UNIVERSITY, KATRA
School of Computer Science & Engineering
B. Tech. (Computer Science & Engineering-Vth Sem.) Major Examination (Odd) 2018-19

Entry No: 2 0 1 6 b c s 0 03

Date: 03.12.2018

Total Number of Pages: [02]

Total Number of Questions: [14]

Course Title: Design & Analysis of Algorithms

Course Code: CSL 3032

Time Allowed: Time Allowed: 3 Hours

Max Marks: [50]

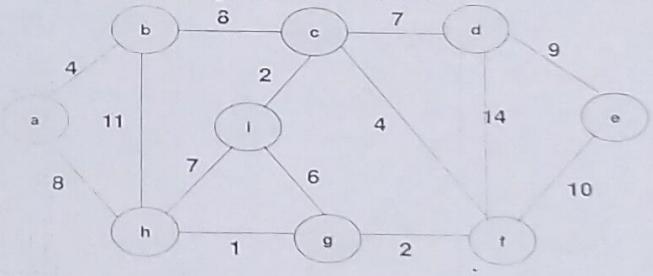
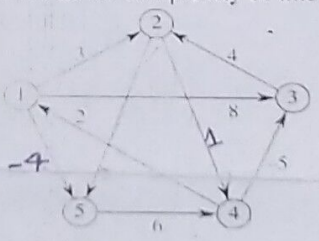
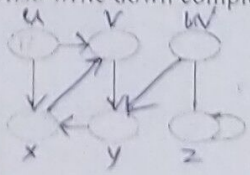
Instructions / NOTE

i. Attempt All Questions.
ii. Support your answer with neat freehand sketches/diagrams, wherever appropriate.
iii. Assume any missing data to suit the case / derivation / answer.
iv. Show every step while solving a numerical/analytical/design based question.
v. Maximum Marks assigned to each question are mentioned with the individual question.

| | Section - A | |
|---|---|---|
| Q1. | Which of the following algorithm design technique is used in finding all pairs of shortest distances in a graph?<br>(A) Dynamic programming<br>(B) Backtracking<br>(C) Greedy<br>(D) Divide and Conquer | [01] |
| Q2. | A list of n strings, each of length n, is sorted into lexicographic order using the merge-sort algorithm. The worst case running time of this computation is<br>(A) O (n log n)<br>(B) O (n2 log n)<br>(C) O (n2 + log n)<br>(D) O (n2) | [01] |
| Q3. | The time factor when determining the efficiency of algorithm is measured by<br>(A) Counting microseconds<br>(B) Counting the number of key operations<br>(C) Counting the number of statements<br>(D) Counting the kilobytes of algorithm | [01] |
| Q4. | The auxiliary space of insertion sort is O(1), what does O(1) mean ?<br>(A) The memory (space) required to process the data is not constant.<br>(B) The amount of extra memory Insertion Sort consumes doesn't depend on the input.<br>(C) It takes only 1 kb of memory<br>(D) It is the speed at which the elements are traversed | [01] |
| Q5. | The number of swapping needed to sort the numbers 8, 22, 7, 9, 31, 19, 5, 13 in ascending order, using bubble sort is<br>(A) 10<br>(B) 9<br>(C) 13<br>(D) 14 | [01] |
| Q6. | What is the significance of Time and Space analysis of Algorithms? | [01] |

| | | |
|---|---|---|
| Q7. | Signifies the Greedy approach used in algorithms. Which algorithms work on the Greedy approach? | [02] |
| Q8. | What is significance of Backtracking? Which algorithms work on backtracking? | [02] |

## Section - B

| | | |
|---|---|---|
| Q9. | What are key differences between 0-1 Knapsack and fractional Knapsack? Prove the statement with appropriate example that "the greedy strategy does not work for the 0-1 knapsack Problem". Also write Complexity of both the knapsacks in all cases. | [06] |
| Q10. | Find out the Minimum Spanning Tree and calculate weight of the same showing the execution of Kruskal's algorithm on the graph mentioned below. Also write down Complexity of the algorithm in all the cases.   | [06] |
| Q11. | Write down John's Algorithm for all-pairs shortest-paths. Show the execution of Johnson's algorithm for computing all-pairs shortest-paths on the following graph. Also write down complexity of this algorithm in all the cases.   | [07] |
| Q12. | Write down the Depth-first-search algorithm and execute the same over the following directed graph considering vertex "u" as source vertex. Show each and every step clearly. Also write down complexity of this algorithm in all the cases.   | [07] |
| Q13. | Write down Floyd-Warshall algorithm. Also note down the complexity of this algorithm in all the cases. Use example | [07] |
| Q14. | What is significance of Divide and Conquer Strategy? Write down Merge Sort Algorithm. Note the complexity of this algorithm in all the cases. Also sort the following numbers in ascending order using merge sort: 5, 2, 4, 7, 1, 3, 2, 6. | [07] |

**SHRI MATA VAISHNO DEVI UNIVERSITY, KATRA**
**School of Computer Science & Engineering**
**B. Tech. (CSE) Minor Examination (Odd/Even/Summer) 2018-19**

**Entry No:** | 1 | 7 | B | C | S | 0 | 4 | 5 |
**Date:**

**Total Number of Pages: [02]**
**Total Number of Questions: [05]**

**Course Title: Design and Analysis of Algorithms**
**Course Code: CSL 3032**

**Time Allowed: 1.5 Hours** **Max Marks: [20]**

Instructions / NOTE (Faculty may include any other relevant instruction, if required)

i. Attempt All Questions.
ii. Support your answer with neat freehand sketches/diagrams, wherever appropriate.
iii. Assume an appropriate data / information, wherever necessary / missing.
iv. Use of IS Code (Mention Number) is permissible in examination.

| | | | |
|---|---|---|---|
| | **Section - A** | | |
| Q1. | (a) Define the following terms:<br>• Computational procedures<br>• Profiling | [02] | CO1 |
| | (b) Suppose program A takes 2n/1000 units of time and Program B takes $1000n^2$ units. For what value of n does program A take less time than program B? | [01] | CO1 |
| | (c) $lg(n!) = O(n \lg n)$ True or False. Justify | [01] | CO2 |
| Q2. | (a) What is Master method for solving recurrence? Define Masters Theorem | [1.5] | CO2 |
| | (b) Find the $\Theta$ notation for $\frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n$ | [1.5] | CO1 |
| | **Section - B** | | |
| Q3. | (a) Solve the following recurrence using recursion tree method<br>$T(n) = T(n/3) + T(2n/3) + n$ | [02] | CO1 |
| | (b) Consider the following recurrence<br>$T(N) = 3T\left(n^{\frac{1}{3}}\right) + \log_3 n$<br>Find its asymptotic bound | [03] | CO1 |
| Q4. | a)Create the tree of calls of MergeSort and Tree of calls of Merge to sort the elements given below:<br>179, 289, 312, 463, 167, 384, 237, 680, 555, 489, 834, 999, 239, 255, 882 | [03] | CO2 |

| | | | |
|---|---|---|---|
| | b) Why is it necessary to have the auxiliary array b[Low:high] in function Merge? Give an example that shows why in-place merging is inefficient. | [02] | CO2 |
| Q5. | Write and Explain Control Abstraction for Divide and Conquer strategy? How is the complexity of divide and conquer algorithms represented in recurrence form. | [03] | CO2 |

Course Outcomes

**CO1** Analyze the run time complexity of algorithms when developed using different approaches like Greedy, Dynamic Programming, Divide and Conquer etc.

**CO2** Identify an appropriate data structure and approach while designing an algorithm for a specific problem.

**CO3** Analytically examine the correctness of algorithms on the basis of recurrence relations, inductive proofs etc.

**CO4** Analyze the Best, Worst and Average Case running time of algorithms and how it is affected by the nature of input variables.

**CO5** Analyze various graph algorithms and deploy these algorithms to model engineering problems.

| CO | Questions Mapping | Total Marks | Total Number of Students (to be appeared in Exam) |
|---|---|---|---|
| CO1 | 1(a), 1(b), 2(b), 3(a), 3(b) | 9.5 | 50 |
| CO2 | 1(c), 2(a), 4(a), 4(b), 5 | 10.5 | 50 |
| CO3 | - | - | |
| CO4 | - | - | |

**SHRI MATA VAISHNO DEVI UNIVERSITY, KATRA**
**School of Computer Science & Engineering**
**B. Tech.(CSE) Minor Examination (Odd/Even/Summer) 2018-19**

**Entry No:** | 1 | 7 | B | C | S | 0 | 4 | 5 | T |

**Total Number of Pages:[02]**

**Date:**

**Total Number of Questions: [05]**

**CourseTitle: Design and Analysis of Algorithms**
**Course Code: CSL 3032**

---

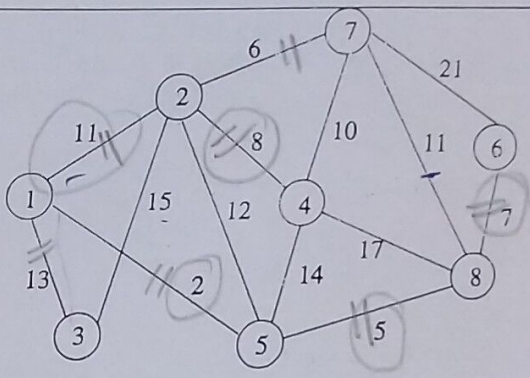
**Time Allowed: 1.5Hours**

**Max Marks: [20]**

Instructions / NOTE(Faculty may include any other relevant instruction, if required)

i. Attempt All Questions.
ii. Support your answer with neat freehand sketches/diagrams, wherever appropriate.
iii. Assume an appropriate data / information, wherever necessary / missing.
iv. Use of IS Code (Mention Number) is permissible in examination.

| | | | |
|---|---|---|---|
| | Section - A | | |
| Q1. | **(a)** What is the prerequisite for Bucket Sort to be effective.<br>**(b)** For solving the Knapsack problem using Greedy approach, considering objects in order of non increasing profit values would yield optimal result. True or Faise. Justify. | [01]<br>[01] | CO1<br>CO2 |
| Q2. | **(a)** Write down the counting sort Algorithm. What are its limitations.<br>**(b)** How can performance of quick sort be improved Explain. What modifications are needed in the basic algorithm of quick sort to enhance its performance. (Write the Modified algorithm) | [3]<br>[3] | CO1<br>CO2 |
| | Section - B | | |
| Q3. | (a) Show how Quick Sort run on the following sequence of keys.<br>10, 15, 3, 18, 45, 35, 17, 19, 57, 90<br>*Do the dry run*. Be explicit with the changes that happen in each iteration. | [02] | CO3 |
| Q4. | **(a)**Discuss Strassens Algorithm for matrix multiplication. Strassens algorithm is computationally better than the conventional matrix multiplication techniques. Justify.<br>**(b)** Find the Minimum cost spanning Tree for the below given graph using Prim's Algorithm. (Do the dry run and create a cost adjaceny matrix. Also show the value of array '**near**' and the cost comparisons being done: for each iteration separately.) | [03]<br>[03] | CO3<br>CO5 |

| | | | |
|---|---|---|---|
| |  | | |
| Q5. | Write a program to sort an array of integers having variable length using Radix Sort | [04] | CO4 |

Course Outcomes

CO1 Analyze the run time complexity of algorithms when developed using different approaches like Greedy, Dynamic Programming, Divide and Conquer etc.

CO2 Identify an appropriate data structure and approach while designing an algorithm for a specific problem.

CO3 Analytically examine the correctness of algorithms on the basis of recurrence relations, inductive proofs etc.

CO4 Analyze the Best, Worst and Average Case running time of algorithms and how it is affected by the nature of input variables.

CO5 Analyze various graph algorithms and deploy these algorithms to model engineering problems.

| CO | Questions Mapping | Total Marks | Total Number of Students (to be appeared in Exam) |
|---|---|---|---|
| CO1 | 1(a) 2(a) | 4 | 50 |
| CO2 | 1(b), 2(b) | 4 | 50 |
| CO3 | 3, 4(a) | 5 | 50 |
| CO4 | 5 | 4 | 50 |
| CO5 | 4(b) | 3 | 50 |

SHRI MATA VAISHNO DEVI UNIVERSITY, KATRA
School of Computer Science & Engineering
B. Tech. (Branch) Major Examination (Even) 2018-19

Entry No: 1 7 B C S 0 4 5
Date:

Total Number of Pages: [07]
Total Number of Questions: [02]

Course Title: Design & Analysis of Algorithms
Course Code: CSL 3032

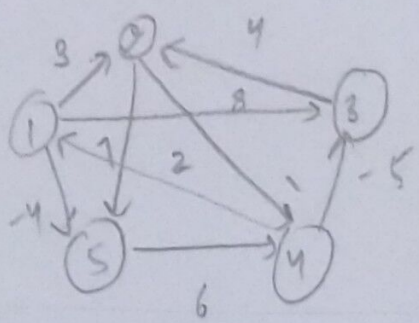
**Time Allowed: 3.0 Hours** **Max Marks: [50]**

Instructions / NOTE

Attempt All Questions.

i. Support your answer with neat freehand sketches/diagrams, wherever appropriate.
ii. Assume an appropriate data / information, wherever necessary / missing.
iii. Use of IS Code is permissible in examination.

| | Section - A | | |
|---|---|---|---|
| Q1. | (a) What are stable sort algorithms? Name any two algorithms which are not stable. | [02] | CO3 |
| | (b) Write and explain the algorithm for generating the Longest Common Subsequence. | [04] | CO3 |
| Q2. | (a) Write the modified Dijkstra algorithm that generates the *shortest path* (in addition to the distance it computes) from a given source vertex to rest of the vertices. | [05] | CO2 |
| | (b) Write and explain the Kruskals algorithm in detail. Explaining each of the functionalities used in it like Heapify, Adjust, Find and Union. | [07] | CO4 |
| | Section - B | | |
| Q3. | (a) Solve the recurrence $T(n) = 2T(n/4) + \sqrt{n}$ by using Masters theorem . | [02] | CO4 |
| | (b) Solve the recurrence $T(n) = T(\sqrt{n}) + 1$ using changing variable method | [03] | CO4 |
| Q4. | Write the code snippet for matrix multiplication of a matrix of order (m x n) and (n x p). Compute its time and space complexity. | [04] | CO4 |
| Q5. | Find the optimal parethesization of a matrix chain product whose sequence of dimensions is <5, 10, 3, 12, 5, 50> | [06] | CO2 |
| Q6. | Run the Floyd Warshall algorithm on the weighted, directed graph of below figure. Show the matrix D(k) that results for each iteration. Also generate the path between every pair of vertices | [08] | CO5 |



| | | | |
|---|---|---|---|
| a) | Let n=4 and (a,b,c,d) = (do, if, int, while). Let p(1:4) = (3,3,1,1) and q(0:4) = (2,3,1,1,1) be the probabilities of success and failure . Compute Optimal Binary Search Tree. | [05] | CO1 |
| b) | Write and explain the Depth first search algorithm for traversing a graph. | [04] | CO5 |

Course Outcomes

CO1Analyze the run time complexity of algorithms when developed using different approaches like Greedy, Dynamic Programming, Divide and Conquer etc.

CO2Identify an appropriate data structure and approach while designing an algorithm for a specific problem.

CO3Analytically examine the correctness of algorithms on the basis of recurrence relations, inductive proofs etc.

CO4Analyze the Best, Worst and Average Case running time of algorithms and how it is affected by the nature of input variables.

CO5Analyze various graph algorithms and deploy these algorithms to model engineering problems.

| CO | Questions Mapping | Total Marks | Total Number of Students (to be appeared in Exam) |
|---|---|---|---|
| CO1 | 7(a) | 5 | 50 |
| CO2 | 2(a), 5 | 11 | 50 |
| CO3 | 1 | 6 | 50 |
| CO4 | 2(b),3,4 | 16 | 50 |
| CO5 | 6, 7(b) | 12 | 50 |